Doc Code: AP.PRE.REQ

PRE-APPEAL BRIEF REQUEST FOR REVIEW

Docket Number (Optional)
T00053

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to "Mail Stop AF, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)]

on ___ July 19, 2005 ___

Signature ___

Typed or printed name ___ Kent B. Chambers

Application Number	Filed
09/902,128	July 10, 2001

First Named Inventor
Ryan Shillington

Art Unit	Examiner
2184	Yolanda L. Wilson

Applicant requests review of the final rejection in the above-identified application. No amendments are being filed with this request.

This request is being filed with a notice of appeal.

The review is requested for the reason(s) stated on the attached sheet(s).
Note: No more than five (5) pages may be provided.

I am the

☐ applicant/inventor.

☐ assignee of record of the entire interest.
See 37 CFR 3.71. Statement under 37 CFR 3.73(b) is enclosed.
(Form PTO/SB/96)

☒ attorney or agent of record. 38,839
Registration number _____

☐ attorney or agent acting under 37 CFR 1.34.

Registration number if acting under 37 CFR 1.34 _____

Signature

Kent B. Chambers
Typed or printed name

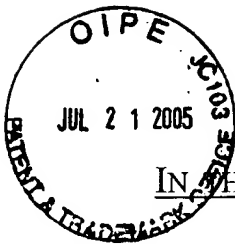(512) 338-9100
Telephone number

July 19, 2005
Date

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required.
Submit multiple forms if more than one signature is required, see below*.

☐ *Total of _____ forms are submitted.

In the UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s):	Ryan Shillington, Will Scott, Dan Burton
Assignee:	Trilogy Development Group, Inc.
Title:	SYSTEM AND METHOD FOR REMOTELY DEBUGGING APPLICATION PROGRAMS
Serial No.:	09/902,128	Filed:	July 10, 2001
Examiner:	Yolanda L. Wilson	Group Art Unit:	2184
Docket No.:	T00053	Customer No.:	33438

Austin, Texas
July 19, 2005

PRE-APPEAL BRIEF ARGUMENTS

Dear Sir:

This paper accompanies the Pre-Appeal Brief Request for Review and sets forth a succinct, concise, and focused set of arguments for which the review is being requested.

CLAIM STATUS

Claims 1-44 are pending.

Claims 1-44 stand rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,668,369, issued to Krebs (referred to herein as "*Krebs*").

ARGUMENTS

Applicants respectfully submit that the Examiner's rejection contains factual errors and, thus, the Examiner has failed to satisfy the PTO's burden of establishing a prima facie rejection.

Krebs teaches that:

> The present invention provides a client-side software debugging tool for viewing dynamic code. Accordingly, the present invention provides a tool for assisting a programmer in locating errors in DHTML and scripts for generating dynamic code. Whereas current debugging tools, such as the "View Source" command of popular Web browser software, permit a programmer to view only

static code as it is received from a Web server, e.g., a script, the present invention permits the programmer to view the dynamic code generated at the client by the script. *Id.*, col. 2, lines 54-63.

The Examiner noted in para. 39 of the Final Office Action that *Krebs* teaches that a "user's computer's web browser software receives a DHTML source file from a server." *Krebs*, col. 4, lines 31-32. Thus, the DHTML source file is initially located on a server and transferred to a user's computer. The Examiner has interpreted the DHTML source file as the 'application program' of the present invention. However, at issue is not **where** the source file was originally located and transferred to, at issue is **where** the application program resides during debugging. Significantly, *Krebs* teaches that all of the debugging takes place using software on the user's computer. The present invention teaches that the application program resides on a server and is invoked/executed while residing on the server during debugging from a workstation. The workstation is remote from the server.

Krebs teaches that all debugging of software takes place using debugging software on the user's computer and the software being debugged also resides on the user's computer. After a **"user's computer's web browser software receives a DHTML source file from a server"**, *Krebs* teaches that

> The source file contains a script for generating dynamic code at the user's computer. **The web browser then runs the script to generate the dynamic code** and stores the dynamic code in internal variables of the browser as known in the prior art, as shown at step 54. The web browser then displays its interpretation of the dynamic code to a user as part of a web page, as is known in the prior art, as shown at step 58. *Id.*, col. 4, lines 32-39.

Thus, *Krebs* teaches that the DHTML source file is run on the user's computer (not the server from which the DHTML was originally transferred) because the DHTML source file has been transferred to the user's computer, and the web browser, which runs the script to generate the dynamic code and store the dynamic code in internal variables, resides on the user's computer.

Krebs also teaches that:

> **The user may then use a software debugging tool in accordance with the present invention as follows. The user first executes the software**

debugging tool, as shown at step 62. This may be achieved in a variety of ways as discussed above. For example, if the debugging tool is integrated into the browser, the user may simply select an appropriate menu option. Alternatively, script commands may be included in the source file to trigger the debugging tool. For example, the "onclick" command could be used to invoke the debugging tool upon the user's depression of the "shift" key and double clicking of a mouse button. The software debugging tool then identifies the dynamic code generated at the user's computer as shown at step 66, e.g., by reference to the variables of the web browser that are used to store the dynamic code, as discussed above. *Id.*, col. 4, lines 39-43.

Thus, *Krebs* teaches that the software debugging tool resides on the same user's computer that the DHTML source file resides on. Accordingly, *Krebs* does clearly teach that <u>all</u> <u>debugging of software takes place using debugging software on the user's computer and the</u> <u>software being debugged also resides on the user's computer</u>. **Applicants respectfully submit that the Examiner made a factual error** in stating that *Krebs* "discloses executing the application program on the server when the application program is invoked from the workstation." Applicants respectfully submit that this error is a material error that undercuts the remaining points of the Examiner's *prima facie* case.

The Examiner stated in the Final Office Action, para. 39, and Applicants agree, that the preambles of the independent claims is to be taken into consideration when interpreting the claims. Each independent claim of the present application relates to "debugging an application program from a workstation, wherein the application program resides on a server that is remote from the workstation." The "workstation" and the "server" are clearly two different computer systems, and the independent claims also explicitly state that the "server [] is remote from the workstation."

The Examiner also states that "*Krebs* does in fact have what is disclosed in the preamble." Final Office Action, para. 39. "The application program is the DHTML source file, that is retrieved from the server, of the webpage seen in the web browser cited in column 4, lines 30-37" of *Krebs*. Final Office Action, para. 39.

However, although the Examiner has identified the DHTML source file originates from a separate computer (server) from the executing computer (the user's computer), Applicants respectfully submit that *Krebs* fails to teach (1) "invoking the application program from the

workstation via a network interface." Claim 1. The Application Program, which "resides on a server that is remote from the workstation", is invoked by the workstation "**via a network interface**." Claim 1. Thus, the workstation is remotely invoking the application program. *Krebs* clearly teaches that the same computer (the user's computer) on which the DHTML source file resides is also the computer that runs the DHTML source file. This is not the case with the present invention, and, thus, *Krebs* does not teach this aspect of the invention. Claim 9 also clearly recites "executing the application program on the server when the application program is invoked from the workstation." Claim 17 recites "means for invoking the application program from the workstation." Claim 21 recites "means for executing the application program on the server when the application program is invoked from the workstation." Claim 29 recites "a user interface operable to allow a user to invoke the application program from the workstation." Claim 36 recites "means for executing the application program on the server when the application program is invoked from the workstation."

Thus, in contrast to the teachings of *Krebs*, the application program of the present invention resides on a server and is invoked from a workstation, wherein the server and workstation are remotely located, and, therefore, separate computers.

As established above, Applicants respectfully submit that *Krebs* clearly teaches that all debugging of software takes place using debugging software on the user's computer and the software being debugged also resides on the user's computer. **In contrast to the teachings of *Krebs***, debugging of the application program as set forth by the claims defining the present invention does not take place on the computer (server) executing the application program, such debugging takes place on a remotely located workstation. For example, claim 1 recites:

> displaying a user frame at the workstation that includes information generated by
> the application program;
> providing a debug view option at the workstation for generating a debug frame of
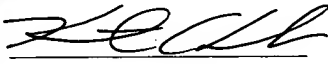> the application program; and
> displaying the debug frame at the workstation when the debug view option is
> selected, wherein the debug frame includes information about one or more
> components of the application program.

Independent Claims 9, 17, 21, 29, and 36 likewise recite debugging operations "at the workstation".

Thus, in summary, *Krebs* teaches that <u>all debugging of software takes place using debugging software on the user's computer and the software being debugged also resides on the user's computer</u> which is in contrast to the claimed invention where the application is invoked/executed from a workstation remotely located from the server on which the application program resides and the debugging set forth in the claims takes place on the remotely located workstation.

Accordingly, Applicants respectfully submit that the Examiner has made a material, factual error in interpreting *Krebs* and accordingly the Examiner has failed to establish a prima facie case.

In light of the above remarks, Applicants respectfully request withdrawal of the rejections.

Respectfully submitted,

Kent B. Chambers
Attorney for Applicant(s)
Reg. No. 38,839